Fully Parallel 30-MHz, 2.5-Mb CAM

Farhad Shafai, Kenneth J. Schultz, *Member, IEEE*, G. F. Randall Gibson, *Member, IEEE*, Armin G. Bluschke, *Member, IEEE*, and David E. Somppi, *Member, IEEE*

***Abstract*—Translation functions in high-speed communications networks such as Internet protocol and asynchronous transfer mode are requiring larger and faster lookup tables. Content addressable memories (CAM's) provide built-in hardware lookup capability with high speed and high flexibility in address allocation. Previous high-capacity CAM's have been inadequate for emerging applications; comparators are time-shared among multiple bits or multiple words, resulting in serialized operation. Fully parallel architectures represent the best solution for high-speed operation, but previous fully parallel CAM's have lacked the capacity required for leading-edge networking applications. This paper describes a fully parallel (single-clock-cycle) CAM chip. The chip uses a 0.35-μm digital CMOS technology to achieve 2.5 Mb of CAM storage and 30-MHz operating frequency. Innovative layout techniques are used to achieve two-dimensional decoding, a traditional problem with high-capacity CAM's. Architecture and operation of the chip are described, including a novel NAND match architecture, operation-specific self-timing loops, and on-board cascade management circuits. The chip functions at 31 MHz, with a search access time of 26 ns and an average search power dissipation of 5.2 W at 25 MHz.**

I. Introduction

In content addressable memories (CAM's), data are accessed based on content rather than physical location. Such lookup capability is useful in many applications, including cache tag tables, translation look-aside buffers, digital neural networks, associative processors, and hardware data base accelerators. Commercially, CAM's have been used most extensively in address translation for local-area-network routing. Recently, similar applications have arisen in high-speed networks, such as gigabit Ethernet and asynchronous transfer mode (ATM); higher speeds and capacities are needed to satisfy requirements of these leading-edge applications. Specifically, our design goal was to satisfy OC-48 (2.5 Gb/s) ATM requirements: sub-100-ns search/read/write operations, and 64-K-entry capacity.

CAM's have continually lagged RAM's in capacity due to the need for integration of comparison circuitry along with bit storage, and also due to the architectural challenge of accomplishing two-dimensional decoding. Various approaches to overcoming these hurdles are described in [1]. Many of these involve time-sharing of comparators among multiple bits or multiple words, resulting in serialized operation. One example of this approach is a bit-serial DRAM-based systolic associative memory [2]. Although serial CAM's have high capacities, they have insufficient speed for today's demanding networking applications. For search rates over 10 MHz, fully parallel architectures represent the best solution but they have relatively low capacity and high power dissipation. The largest fully parallel CAM (with volatile storage elements) reported to date has a capacity of only 336 Kb [3] and a power-performance metric of 270 fJ/bit/search.

Manuscript received April 3, 1998; revised June 8, 1998.

The authors are with Nortel Semiconductors, Nortel (Northern Telecom), Ottawa, Ont. K1Y 4H7 Canada.

Publisher Item Identifier S 0018-9200(98)07034-6.

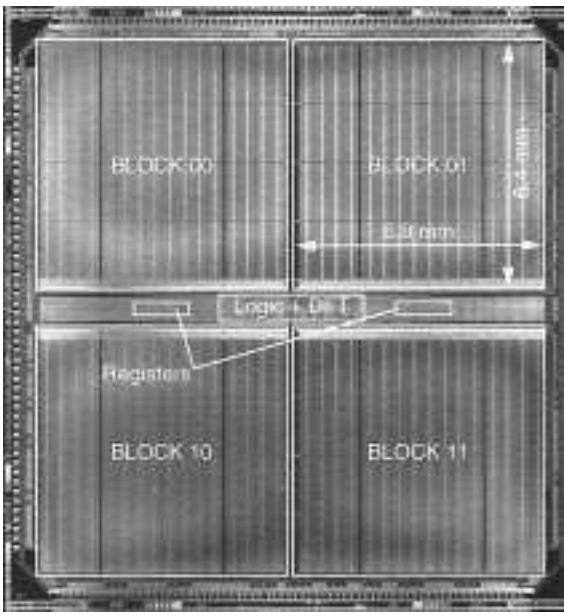

Fig. 1. Die photo of the chip, which measures 15.25 by 15.25 mm. The 64-K-entry address space is divided into four 16-K blocks; each block is 6.8 mm wide and 6.4 mm high. Logic and DFT circuits are placed in the horizontal space between the upper and lower blocks.

In recent years, research focus appears to have shifted to nonvolatile CAM's [4]–[7], with some of these having either multivalued [5], [6] or analog [7] storage elements. Aside from [6], these all have search times in excess of 100 ns and all have long program/erase cycles. The capacities vary from 1 to 1.28 Mb. Therefore, none meets our application requirements.

In this paper, a 30-MHz, 64 K$\times$ 40 CAM with a power-performance metric of 83 fJ/bit/search is presented. Section II describes the architecture of the chip, while Section III details the design of a single block, including the NAND-based match function and the operation-specific self-timing loops. The chip cascading capability is the subject of Section IV. Experimental results and conclusions are presented in Sections V and VI, respectively.

II. Chip Architecture

A die photo of the device is shown in Fig. 1. In addition to the 64 K$\times$ 40 CAM array, the chip contains cascade management logic, interface logic, and design-for-test (DFT) features. The DFT features include Scan, ScanBIST, and a

0018–9200/98$10.00 © 1998 IEEE

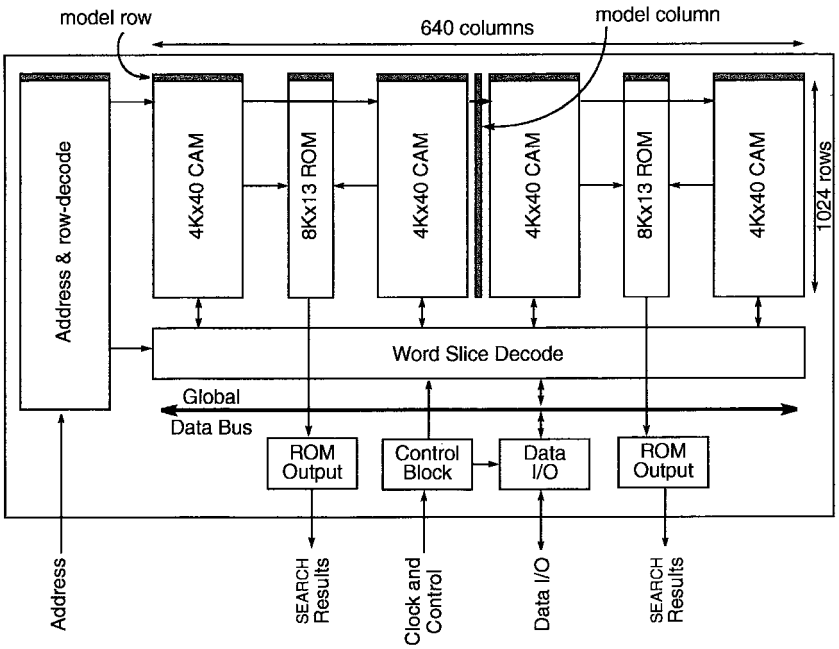

Fig. 2. Block diagram of a block.

CAM BIST controller, which is run at-speed with a custom algorithm based on real physical fault models. The CAM has a clocked architecture, similar to [8]. All operations are self-timed from the rising edge of CK using operation-specific self-timing loops, and all inputs must satisfy setup and hold requirements around this clock edge. Because all inputs are latched internally, setup and hold times are small. The device draws near-zero (leakage only) standby current (CK inactive); otherwise, power dissipation is proportional to the frequency of active clock cycles.

To bring the performance level up to the required specifications, the 64-K-entry address space is divided into four 16-K blocks. Each block is an independent entity capable of performing search, read, and write operations with minimal external control circuitry. In a search operation, the four blocks operate concurrently, with each searching one-quarter of the address space. In read or write operations, only one of the four blocks is active.

Having four separate blocks requires additional logic to perform address decoding in read and write operations and output multiplexing in search operations. The blocks are designated by the two most significant address bits. The logic circuitry decodes these two bits and selects one of the blocks in read and write operations. In search operations, the interface logic combines the outputs of all four blocks onto a common output data bus. Combining the four individual search results involves generating chip hit and multiple match signals, generating the two most significant address bits, and multiplexing the appropriate block outputs onto the output data bus.

III. Block Design

A. Block Architecture

A block, shown in Fig. 2, is 6.4 mm high$\times$ 6.8 mm wide, with 1024 rows and 640 columns. A vertical stack of 1024 words, along with its column circuits, is referred to as a word slice. There are 16 word slices across a block—four sets of four, with each set shown as a 4 K by 40 sub-block in Fig. 2; a pair of sets is associated with each ROM.

Ignoring the two ROM's and their corresponding output circuitry, the block reduces to a standard memory array with CAM core cells. The row decoding is performed by ten x-address bits, and the word-slice decoding is done by four z-address bits. A 40-bit bidirectional global data bus is used to transfer data between the I/O interface and the 16 word slices, as shown in Fig. 2.

The ROM's are used to encode the result of a search operation. After a search operation is initiated, the search data, or comparand, are applied via the global data bus to all 16 word slices. If the search data match the data stored in any word, a match line, unique to that word, is asserted. The match lines control access to the two ROM's, effectively functioning as ROM word lines. The data stored at each ROM location correspond to the binary address of the associated CAM word, and hence the accessed ROM word stores the address of the CAM word that matched the comparand. Each ROM encodes half of the block's address space, and the SEARCH result is obtained by reading the ROM's when a match line is asserted.

Each ROM is logically 8 K words by 13 bits. The three least significant bits correspond to the address of the word slices in a row (there are eight word slices associated with each ROM per row). Since encoding these three bits requires the same logic for every row, they are implemented as eight unencoded bit lines, and the encoding logic is instantiated only once: in the ROM output circuitry. The ten most significant bits have both true and compliment bit lines, as in [3]. If one polarity is pulled down for any bit, a match has occurred. If both polarities are pulled down for any bit, a multiple match has occurred. Each ROM outputs a "hit," "mult," and search

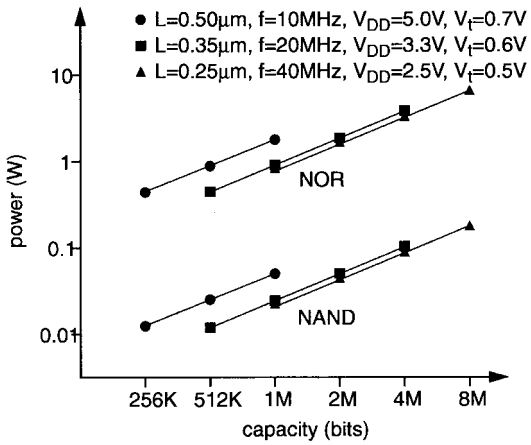

Fig. 3. Match-line power for NOR and NAND architectures.

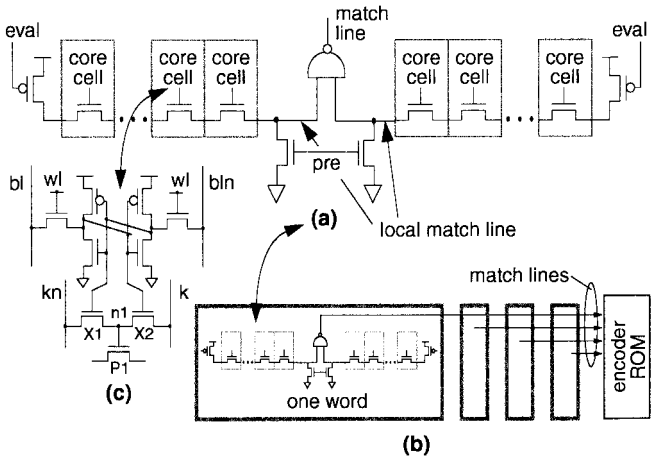

Fig. 4. NAND match chains: (a) one word, (b) four words, and (c) core cell.

address "sa." The outputs of the two ROM's are combined externally to produce the block's search results.

Because of the long wire lengths, all global signals are routed in upper metal layers, many in metal5, and their resistance and capacitance are modeled in simulations. Significant delays are incurred in transit across the block, so timing control signals a) are generated in a sequence that avoids race conditions, b) travel in the same direction as the signals they are qualifying, and c) are locally reset when this tracking is not possible. To model transit delays, the self-timing loops utilize a model column in the middle of the array and a model row at the top of the array.

B. CAM Word

One of the fundamental challenges in designing high-capacity CAM's is minimizing the power associated with match line transitions. In a fully parallel architecture, all stored bits are compared to the search data in one clock cycle. In traditional implementations [3], the result of a word search is determined by a dynamic NOR of the bit comparison results. This leads to two match-line transitions per mismatched word per search. In our implementation, the dynamic NOR function is replaced by a dynamic NAND chain. This results in two match line transitions per matched word per search. In a typical system, the number of misses is expected to be much greater than the number of matches; thus, using a dynamic NAND structure results in a significant reduction in power. Fig. 3 demonstrates the power advantages of a NANd architecture versus a NOR.[1]

A NAND match-line architecture is considerably slower than its NOR counterpart, especially for wide words. To improve NANd chain speed, each 40-bit word is divided into two 20-bit chains. The two local match lines (the outputs of each half) are NAND-ed to produce the match line as shown in Fig. 4(a). The resulting match line is routed to a ROM. Fig. 4(b) shows four words in one row on one side of a ROM. Four match lines run in metal4 over each core cell. This arrangement is mirrored to the right of the ROM. Recall the traditional CAM architectural hurdle of two-dimensional decoding; here, this is overcome by exploiting an upper metal layer to allow the placement of adjacent words in a single row.

The core cell is shown in Fig. 4(c). It comprises a standard 6-T SRAM cell,[2] two XOR transistors (X1 and X2) used for comparison, and a pass transistor (P1) that is part of the NAND chain. In a search operation, the search data are differentially driven onto the compare lines k/kn. For example, if the cell is storing a "1" value (X2 is on) and the search is for a "1," k and kn are driven high and low, respectively, and as a result, node n1 is charged high through X2 and P1 is turned on. If the search is for a "0," k is driven low, n1 is discharged, and P1 is turned off.

The word line wl runs horizontally in metal2, while the bit lines bl/bln and compare lines k/kn run vertically in metal3. Because either X1 or X2 is conducting in each cell, the load on k/kn varies with data stored in the cells of a column.

To avoid charge sharing between the intermediate nodes of the dynamic NAND chain and the local match line, the intermediate nodes must be precharged to the same voltage as the local match lines. It is difficult to determine a proper "high" precharge value for the intermediate nodes because the NAND stack is composed of back-biased NMOS transistors. Therefore, a precharge level of ground was selected; evaluation is upward. The precharge of the intermediate nodes is accomplished by setting the quiescent state of k/kn high. With both k and kn high, all pass transistors are turned on irrespective of the data stored in the cell. Therefore, the precharge of all dynamic nodes occurs after the active part of the cycle has ended and k/kn have returned to their quiescent state.

The high quiescent state for k/kn results in the P1 transistors of all cells in the entire array's conducting at the beginning of the subsequent search operation. Therefore, match evaluation is independent of the comparand, and the slowest possible result is a match; there is no concern about variable-speed results, as in a NOR-based CAM. However, to avoid false

[1] The graph assumes × 64 configurations, two NAND chains per word, and a statistical average of approximately two adjacent matched bits needing precharging per half-word per search. The signal swing in the NAND chain is assumed to be VDD-Vt, with the Vt's as shown.

[2] Although nonvolatile storage is more suited to high-capacity CAM's, the design was implemented in a digital application-specific integrated-circuit (ASIC) process technology to allow use of the CAM blocks as embedded macros in other ASIC's, avoid the extra cost of nonvolatile process technology, and achieve single-cycle write operations.

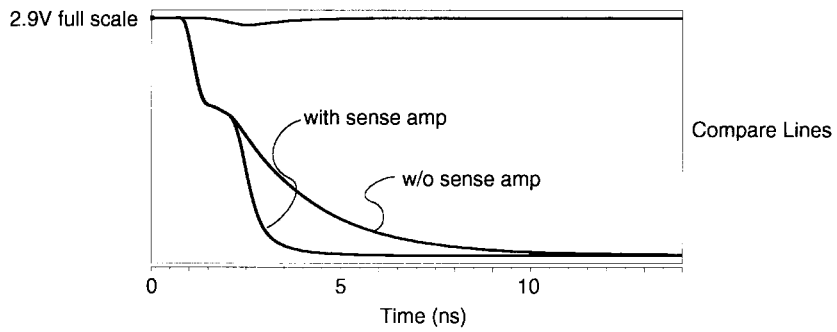

Fig. 5. Voltage (at 125°C) of the active compare line in the model column at the top of the block with and without a sense amplifier.

matches, it is vital for match evaluation, triggered by a falling transition on "eval," to start after the comparand has been driven onto k/kn and the P1 transistors have had a chance to fully turn off.

C. Improving Compare-Line Drive

As mentioned above, the load on k/kn is data dependent and could be very high if all of the cells are storing the same polarity. To compound the problem, the compare lines run the full height of the block (6.4 mm) and are quite resistive (500 Ω). This results in a large resistance–capacitance (RC) time constant and limits how fast data can be driven onto the compare lines.

To overcome this problem, a sense amplifier is connected to each of the compare lines at the top of the block. This sense amplifier, when clocked, senses the polarity of the data being driven onto the compare lines and turns on an additional driver. Having drivers at the top and bottom of the array effectively reduces the RC effects by a factor of four. Fig. 5 shows simulation waveforms[3] of compare-line voltage at the top of the block with and without a sense amplifier.

D. Search Self-Timing Loop

Fig. 6 shows the major components of the search self-timing loop. It shows one of the 16 word slices, one of the ROM's, the model column, the control block, I/O drivers, and the logic used in generating control signals. The model column has one active compare line knm, with the other compare line km permanently set high. The core cells in this column are programmed to maximally load knm.

The model global data-bus signal dgm has the same loads as the real data bus but known transition directions. It is pulled high at the same time as the input drivers are turned on at the beginning of an operation and is used to qualify the compare-line drivers to avoid power-consuming glitches on k/kn. When the self-timing loop completes, dgm is pulled low and the control logic is reset.

The sequence of events in a search operation is as follows (the signal names are defined in the figure).

1) Following the rising edge of the system clock ck, the control block asserts cd, resulting in the following: a) the input data drivers are enabled and the comparand is driven onto the global data bus, b) dgm is pulled high, and c) the compare-line drivers in each column are qualified via cs.

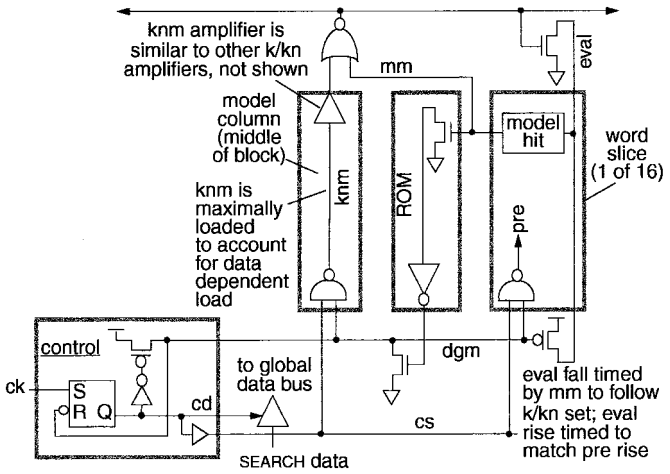

Fig. 6. Search control-signal generation.

2) When dgm becomes valid, the pullup device on eval is turned off, the compare-line drivers are enabled, and the precharge of the local match lines is disabled via the pre signal. The pre signal travels the full height of the block and is used to clock the k/kn sense amplifiers. Using pre as the sense-amplifier clock guarantees sufficient differential on k/kn and correct sense-amplifier operation.

3) The active compare line in the model column knm is monitored at the top of the array, and when it reaches a sufficiently low voltage, the eval signal is asserted low and match evaluation is initiated. Since knm is maximally loaded, using it to assert eval guarantees that all compare lines are valid by the time match evaluation is started; false matches are thus eliminated.

4) The self-timing loop uses a dummy word in the model row that is programmed to hit irrespective of the search data. Since the slowest possible result is a match, the model match line mm associated with the dummy word in the model row is asserted at about the same time as other match lines. When mm is asserted, a) the pulldown device on eval is turned off, b) a dummy bit line in one of the ROM's is discharged, and c) although not shown

[3]The internal nodes of the device are blocked by a top-metal power grid. Therefore, only simulation waveforms of internal nodes are presented in this manuscript.

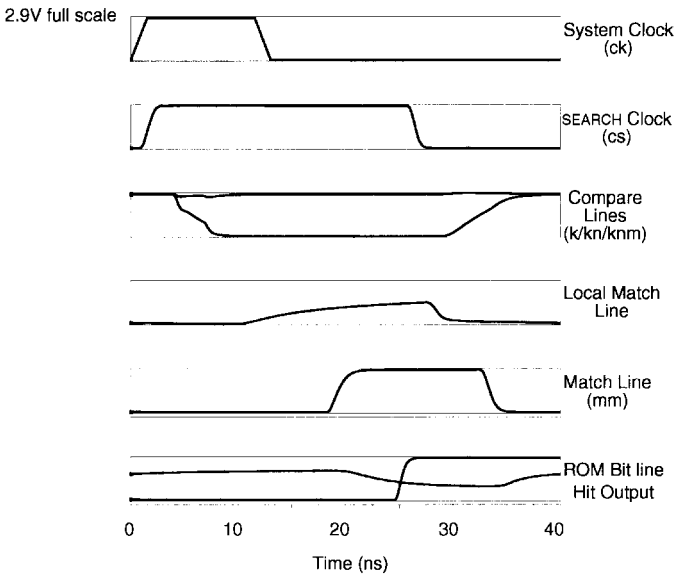

Fig. 7. Simulated search waveforms.

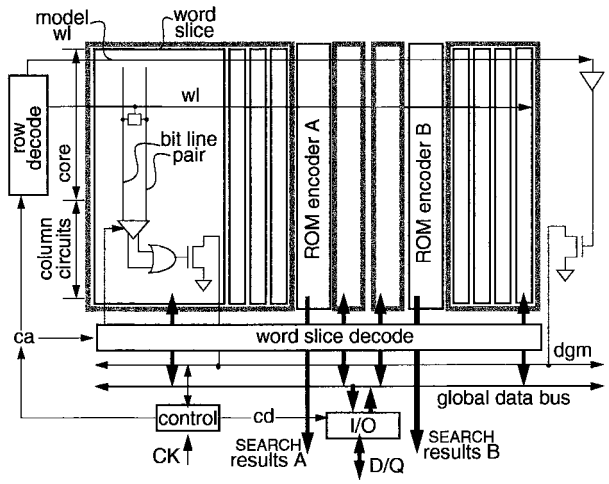

Fig. 8. Read/write control-signal generation.

in the figure, the k/kn sense amplifiers and drivers at the top of the block are turned off.

5) When the ROM control block senses the transition of the dummy bit line, dgm is pulled low. The dummy bit line has the same loads as other ROM bit lines. Therefore, the time it takes for it to be sensed is the same as other ROM bit lines.
6) The falling transition of dgm signals completion of the self-timing loop; at this time, the precharge of the local match lines begins, the eval signal is deasserted, the k/kn drivers are turned off, and the control block is reset.

Fig. 7 shows simulation waveforms of these events.

E. Read and Write Self-Timing Loops

Fig. 8 shows a block diagram of both read and write timing loops. As in a search operation, a rising transition on dgm occurs at the beginning of a cycle and its falling edge indicates the completion of the timing loop. In both read and write operations, the control block clocks the row and word-slice decoders by asserting the address clock ca.

In a READ operation, the output of the sense amplifier of one of the bits in the active word slice is monitored. As soon as a "1" or "0" is sensed, dgm is pulled low and the cycle ends. Since dgm has the same load as the global data bus, the time required for dgm to be pulled low is similar to the time required to drive the output data onto the global data bus.

In a write operation, the data input clock cd is asserted and data are driven onto the global data bus. The write drivers are gated by dgm to ensure they do not turn on until the global data bus is valid. The write timing loop uses the model row at the top of the array. The word line in this row is always active in write operations. It simply travels the width of the array, gets buffered, and goes back to the bottom of the block, where it pulls down dgm. The write cycle has so much margin that using this simple timing loop does not pose any problems.

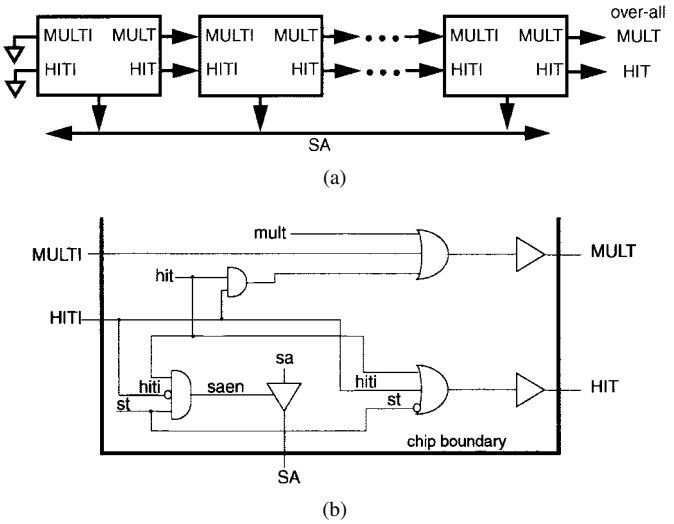

Fig. 9. Cascade management logic. (a) Typical cascade configuration. (b) On-board cascade management circuits.

IV. Cascading in Depth

For systems requiring multiples of 64-K entries, an unlimited number of devices can be cascaded in depth as shown in Fig. 9(a). In this configuration, devices to the left have higher priority than those to the right. The search results are driven onto a common external bus SA, and the overall system HIT and MULT signals are obtained by monitoring the HIT and MULT outputs of the lowest priority chip.

Each device has on-board cascade management logic (similar to [9]), shown in Fig. 9(b), to eliminate contention on the search address output bus (SA). The management logic uses st, a simple inversion of dgm from one of the blocks, to disable the SA drivers, via saen, and assert the HIT output at the beginning of a cycle. If the highest priority device has a match, its HIT signal remains high and the SA drive of all lower priority chips remains disabled. However, if the first device misses, its HIT output goes low and this transition is cascaded through the chips until one has a match. If none of the devices has a match, the overall HIT signal eventually goes low.

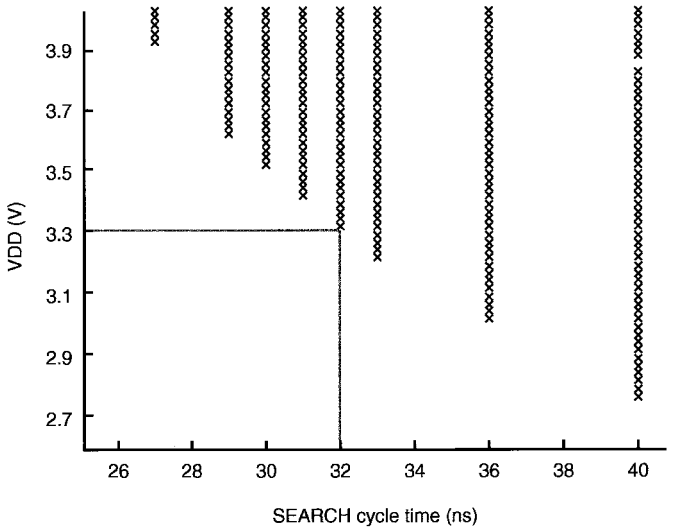

Fig. 10. Shmoo plot showing voltage versus search cycle times. The ×'s show passing regions. All numbers are measured at 25°C.

If there are n devices in the cascade, then the maximum SA access time is

$$\mathrm{SA}t_{\mathrm{ACC}} = t_{\mathrm{SRCH}} + (n-2)t_{\mathrm{HIH}} + t_{\mathrm{HISA}}$$

where t_{SRCH} is the time it takes for a device to search its contents, t_{HIH} is the propagation time of HITI to HIT in a device, and t_{HISA} is the HITI-to-SA access time of the last device. Thus, unlimited cascading is achieved with a trivial interface and deterministic performance.

V. Results

The chip was fabricated in a 0.35-μm digital CMOS process technology with five metal layers and one polysilicon layer. The chip is 15.25 by 15.25 mm with more than 26 million transistors. At 3.3 V and 25°C, the measured cycle time is 32 ns, and search access time is 26 ns. The HITI to HIT propagation time (t_{HIH}) and HITI-to-SA access time (t_{HISA}), at the same operating conditions, are both 5.5 ns. Hence, for a 512-K-entry system ($n = 8$), the maximum SA access time is 64.5 ns.

Average power dissipation at 3.3 V and 25°C, while performing only search operations at 25 MHz, is 5.2 W. In a typical system application with search, write, read, and unselected cycles, the average power dissipation can be expected to be less than 4.0 W at 25 MHz.

The shmoo plot in Fig. 10 depicts the dependency of search cycle time on voltage measured at 25°C. Fig. 11 shows waveforms of the system clock HIT and two bits of the SA bus for two consecutive search operations at 10 MHz. The first cycle results in a match. In this case, the HIT signal remains high and the SA signals have 26-ns access times. The second cycle results in a miss, causing the HIT signal to go low after 26 ns and tristating the SA bus drivers.

Device characteristics are summarized in Table I.

VI. Conclusion

In this paper, we presented a CAM designed to satisfy OC-48 ATM requirements. The high performance is achieved by dividing the 64-K-entry address space into four blocks functioning in parallel and by employing architectural and circuit innovations such as operation-specific self-timing loops, two NAND chains for match evaluation, and multiple compare-line drivers. The novel architecture and design result in search access time of 26 ns and cycle time of 32 ns.

The chip dissipates an average power of 5.2 W when performing search operations at 25 MHz. The low power is achieved using a NAND match architecture and a robust search self-timing loop. The NAND match architecture generates match-line transitions only for words that match the comparand. The self-timing loop models real signal delay across a block and sequences control-signal transitions to minimize crowbar currents.

The chip has on-board cascade management circuits and can be cascaded in depth, with a simple interface, for systems requiring multiples of 64-K entries.

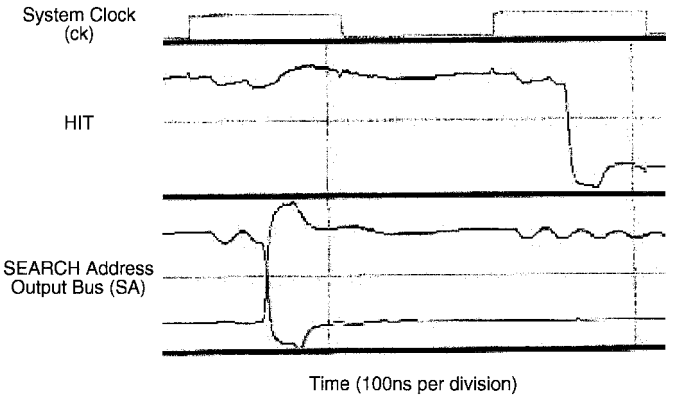

Fig. 11. Waveforms showing the system clock HIT and two bits of the SA bus. The clock period is 100 ns. The first cycle is a search resulting in a match. In this case, the HIT signal remains high and the search address outputs have access times of 26 ns. The second cycle results in a miss, causing the HIT signal to go low after 26 ns and tristating the SA bus drivers.

TABLE I
Chip Characteristics. Parametric Numbers Are Measured at 3.3 V and 25°C

Chip configuration	64K x 40
Block configuration	4 blocks of 16K x 40
Die size	15.25mm x 15.25mm
Package	208-pin PQFP w. slug
Number of transistors	26M
Cycle Time	32ns
SEARCH access time	26ns
HITI to HIT propagation delay	5.5ns
HITI to SA access time	5.5ns
Average power dissipation	5.2W
Power Performance	83fJ/bit/SEARCH
Process technology	0.35µm, 5 metal, 1 poly
Core cell size	9.1µm x 5.9µm

Acknowledgment

The authors wish to thank all who contributed to the development of this chip, notably R. Alleyne, S. Gupta, P. Reside, K. Pagiamtzis, J. Podaima, B. Maritz, S. Adham, F.

Black, T. Dillon, G. Marcantonio, L. Sasaki, G. Fortier, R. Schute, L. Brabant, and I. Martin.

REFERENCES

[1] K. J. Schultz and P. G. Gulak, "Architectures for large-capacity CAM's," *Integration: VLSI J.*, vol. 18, no. 2/3, pp. 151–171, 1995.
[2] G. J. Lipovski, "A four megabit dynamic systolic associative memory chip," *J. VLSI Signal Process.*, vol. 4, pp. 37–51, Feb. 1992.
[3] T. Ogura, M. Nakanishi, T. Baba, Y. Nakabayashi, and R. Kasai, "A 336-Kbit content addressable memory for highly parallel image processing," in *Proc. IEEE Custom Integrated Circuits Conf.*, 1996, pp. 273–276.
[4] T. Miwa, H. Yamada, Y. Hirota, T. Satoh, and H. Hara, "A 1 Mb 2-transistor/bit nonvolatile CAM based on flash-memory technologies," in *ISSCC Dig. Tech. Papers*, 1996, pp. 40–41.
[5] T. Hanyu, N. Kanagawa, and M. Kameyama, "One-transistor-cell multiple-valued CAM for a collision detection VLSI processor," in *ISSCC Dig. Tech. Papers*, 1996, pp. 264–265.
[6] T. Hanyu, M. Arakaki, and M. Kameyama, "2-transistor-cell 4-valued universal-literal CAM for a cellular logic image processor," in *ISSCC Dig. Tech. Papers*, 1997, pp. 46–47.
[7] A. Kramer, R. Canegallo, M. Chinosi, D. Doise, G. Gozzini, L. Navoni, P. L. Rolandi, and M. Sabatini, "55GCPS CAM using 5 b analog flash," in *ISSCC Dig. Tech. Papers*, 1997, pp. 44–45.
[8] A. L. Silburt, R. S. Phillips, G. F. R. Gibson, S. W. Wood, A. G. Bluschke, J. S. Fujimoto, S. P. Kornachuk, B. Nadeau-Dostie, R. K. Verma, and P. M. J. Diedrich, "A 180-MHz 0.8-μm BiCMOS modular memory family of DRAM and multiport SRAM," *IEEE J. Solid-State Circuits*, vol. 28, pp. 222–232, Mar. 1993.
[9] T. Ogura, S. Yamada, and T. Nikaido, "A 4-Kbit associative memory LSI," *IEEE J. Solid-State Circuits*, vol. SC-20, pp. 1277–1282, Dec. 1985.

Farhad Shafai was born in Tehran, Iran, in 1970. He received the B.A.Sc. degree in electrical engineering from the University of Toronto, Canada, in 1995.

He joined the Memory Development Group at Nortel (Northern Telecom), where he has been responsible for content addressable memory, custom embedded memory, and application-specific integrated-circuit flow development.

Kenneth J. Schultz (S'85–M'96) was born in Winnipeg, Man., Canada in 1965. He received the B.Sc. degree (Gold Medalist) from the University of Manitoba in 1987 and the M.A.Sc. and Ph.D. degrees from the University of Toronto, Canada, in 1989 and 1996, respectively, all in electrical engineering.

From 1989 to 1991, he was with the Memory Development Group at Northern Telecom/BNR, Ottawa, Ont., Canada. After completing his doctoral work on content addressable memory (CAM)-based circuits for asynchronous transfer mode switching in 1995, he returned to the group. He currently has responsibilities in the areas of CAM development and Internet protocol reuse. His research interests are in the area of novel memories for telecom and datacom applications.

Dr. Schultz received the Natural Sciences and Engineering Research Council of Canada 1967 Scholarship and the 1994–1995 IEEE Solid-State Circuits Council Predoctoral Fellowship. He was a corecipient of the Best Paper Award at the Canadian Conference on VLSI in 1988 and 1993. He is a licensed Professional Engineer and is currently Chair of the IEEE Ottawa ED/CAS/SSC Chapter.

G. F. Randall Gibson (S'85–M'88) was born in Ottawa, Ont., Canada, in 1964. He received the B.Eng. degree in electrical engineering from Carleton University, Ottawa, in 1987.

He joined the design analysis group at MOSAID, Inc., Carp, Ont., in 1987, where he was involved in the design and failure analysis of SRAM, DRAM, and RAMDAC products. In 1989, he joined the memory development group at Bell-Northern Research, now Nortel (Northern Telecom), Ottawa, where he has been involved in the development of high-speed and low-power embedded memory, including content addressable memory (CAM), DRAM, multiport SRAM, ROM, and nonvolatile memory. He now manages the Memory Development Group at Nortel (Northern Telecom). He has published several papers. He has received four patents with five patents pending in the fields of embedded memory, multiport memory test, and CAM.

Armin G. Bluschke (M'82) was born in Hamilton, Canada, in 1959. He received the B.Eng. degree from McMaster University, Hamilton, in 1981 and the M.S. degree from the University of California, Berkeley, in 1986.

From 1981 to 1984 and 1986 to 1990, he was with Gennum Corp., Burlington, Canada, designing low-voltage bipolar amplifiers. From 1990 to 1995, he was with Bell-Northern Research, Ottawa, Ont., Canada, designing high-speed CMOS and BiCMOS embedded memories. Since 1995, he has been an adviser to the COT development group at Nortel (Northern Telecom), Ottawa.

David E. Somppi (M'98) received the B.A.Sc. and M.A.Sc. degrees in electrical engineering from the University of Waterloo, Canada, in 1980 and 1984, respectively.

He was a Design Engineer with Gennum Corp. from 1980 to 1988, concentrating on low-power analog and radio-frequency design. He has been with Nortel (Northern Telecom) since 1988, where he has had technical and management responsibilities in both silicon and system-level product groups. He is presently Product Line Manager, Datacom Products, Nortel Semiconductors, Ottawa, Ont., Canada.